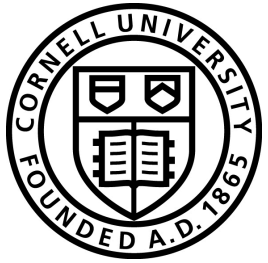
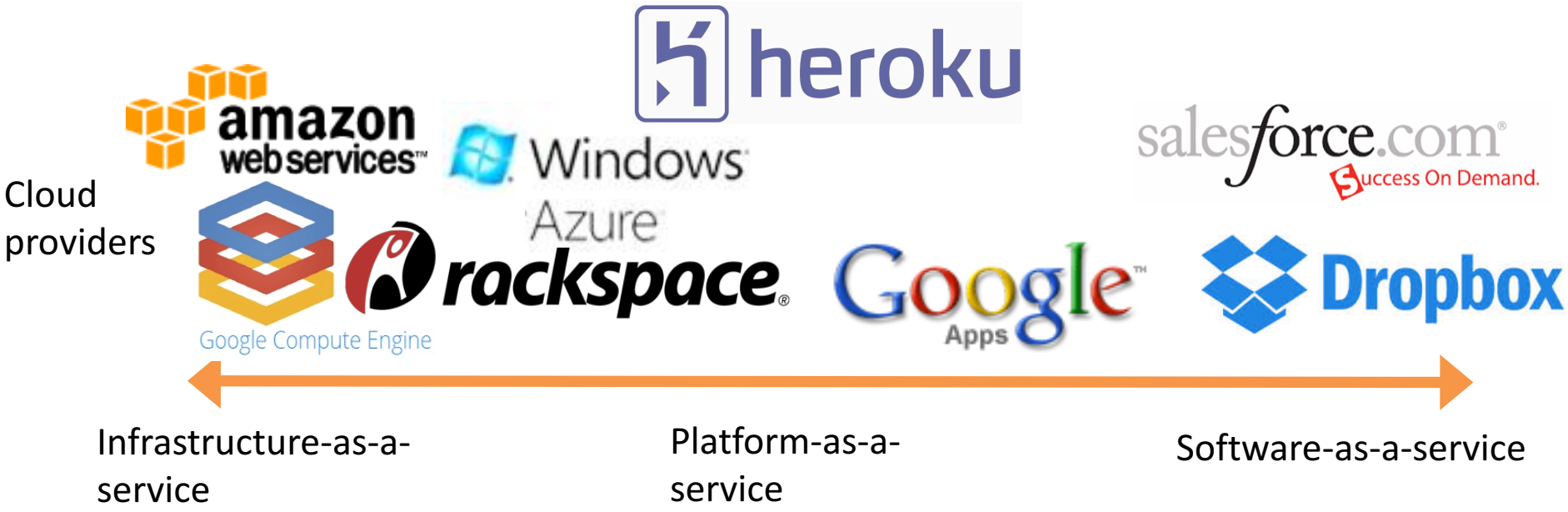
Cloud security

Tom Ristenpart
CS 6431



Cloud computing

NIST: Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.



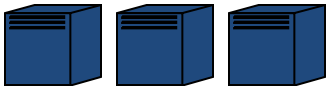
“If computers of the kind I have advocated become the computers of the future, then computing may someday be organized as a public utility just as the telephone system is a public utility... The computer utility could become the basis of a new and important industry.”

— John McCarthy, speaking at the MIT Centennial in 1961

Public IaaS clouds (Infrastructure-as-a-Service)

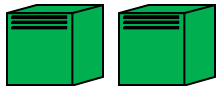
User A

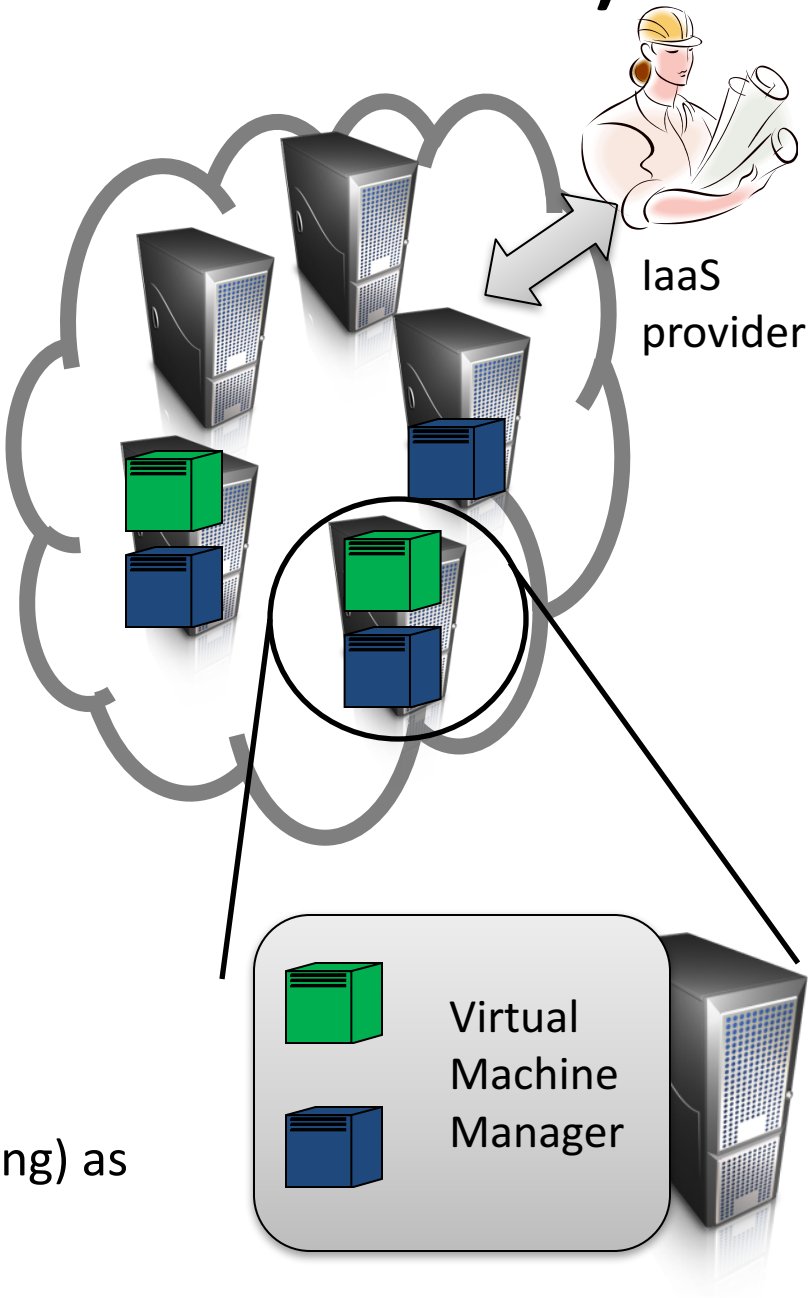
virtual machines (VMs)



User B

virtual machines (VMs)





User's pay for resources used

Multitenancy (users share physical resources)

Virtualization of resources (compute, networking) as managed by provider

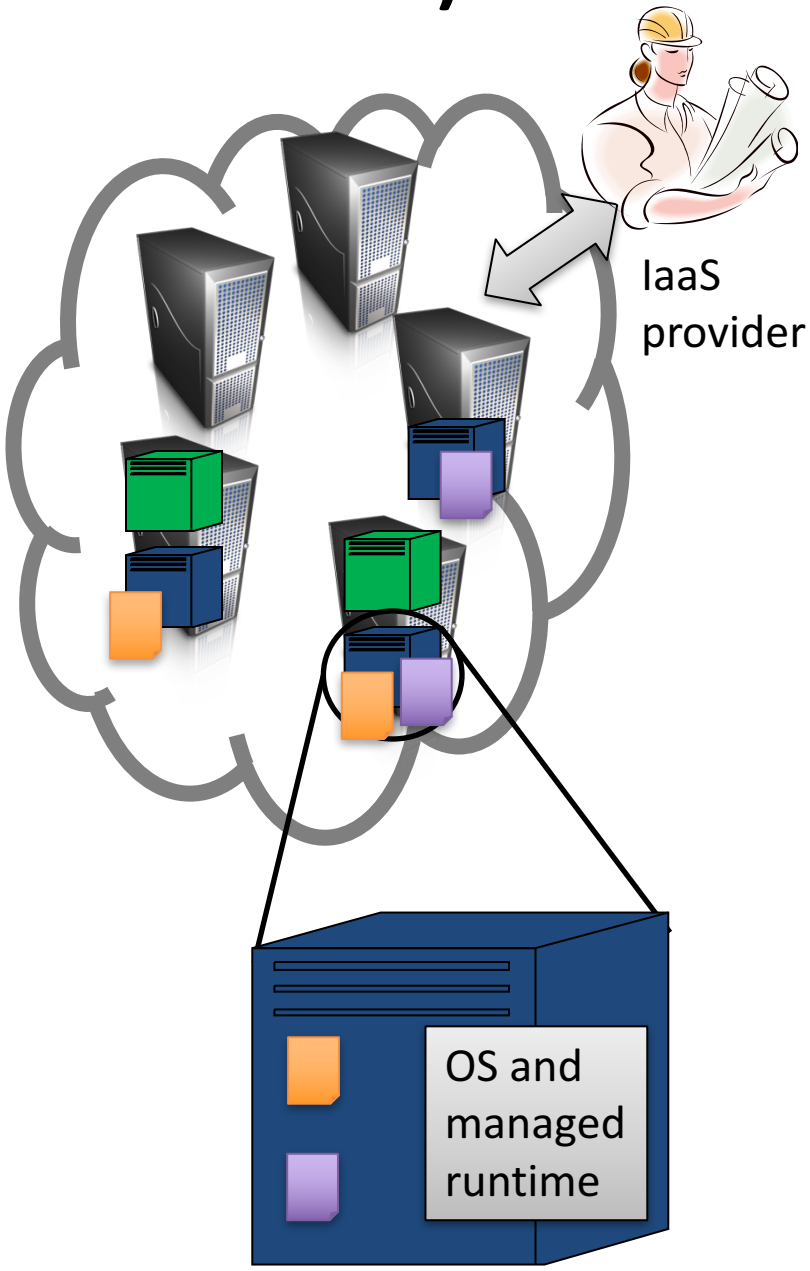
Public PaaS clouds (Platform-as-a-Service)

PaaS user 1

Application programs

PaaS user 2

Application programs

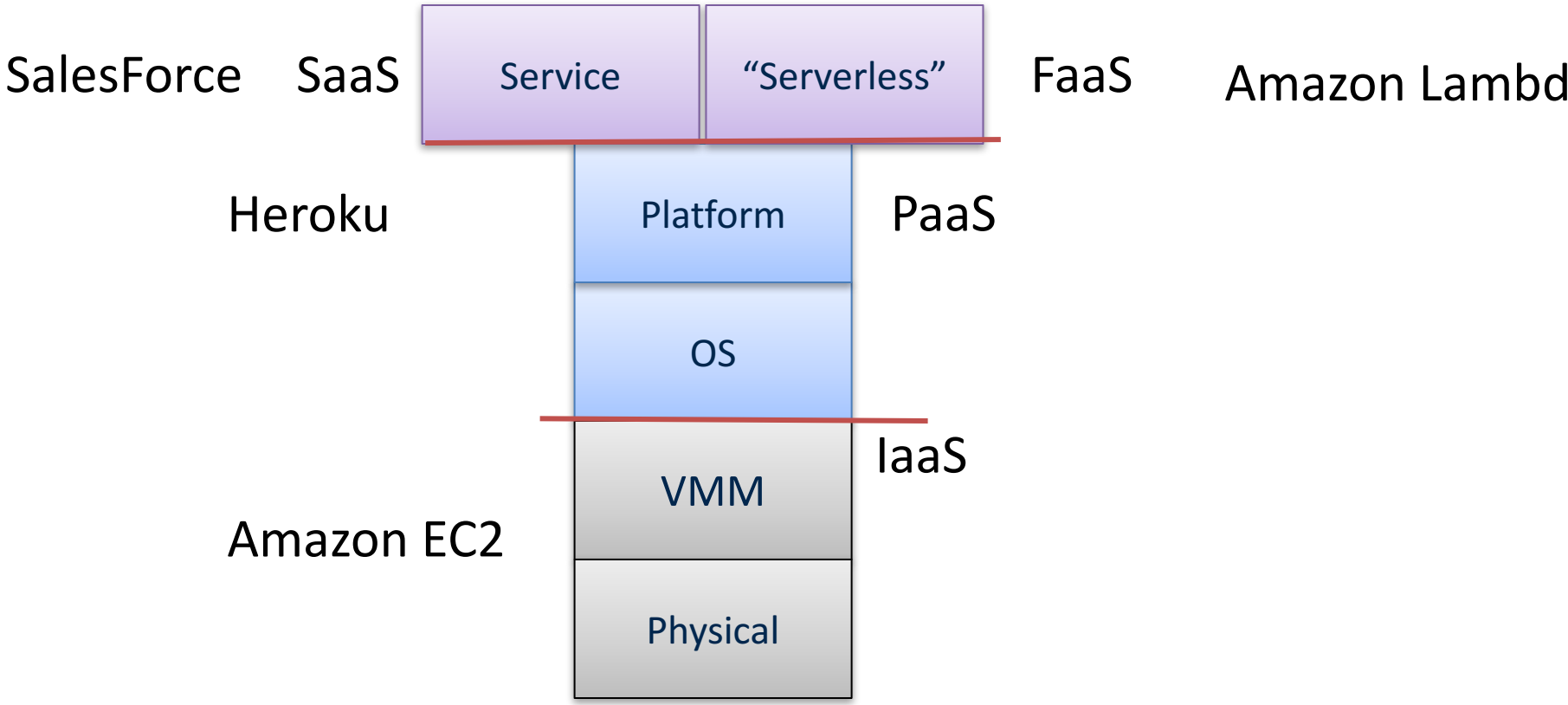


PaaS users submit applications:

- Docker containers (process groups)
- PHP / Python / Java / Ruby apps
- MapReduce / Spark programs

PaaS provider manages runtime

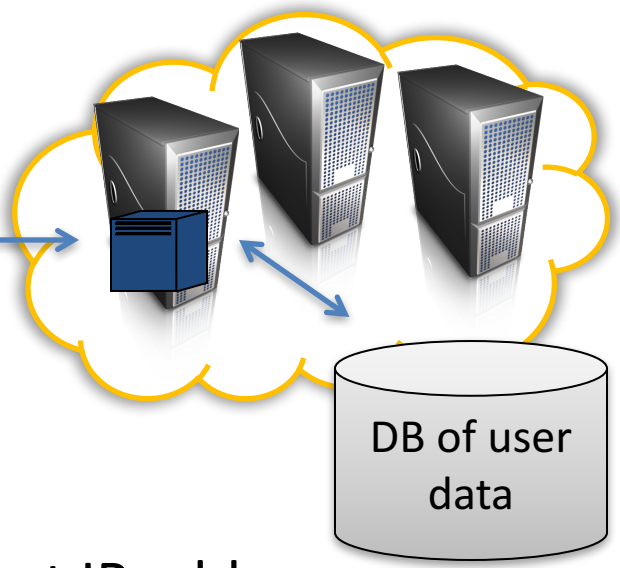
A layered ecosystem



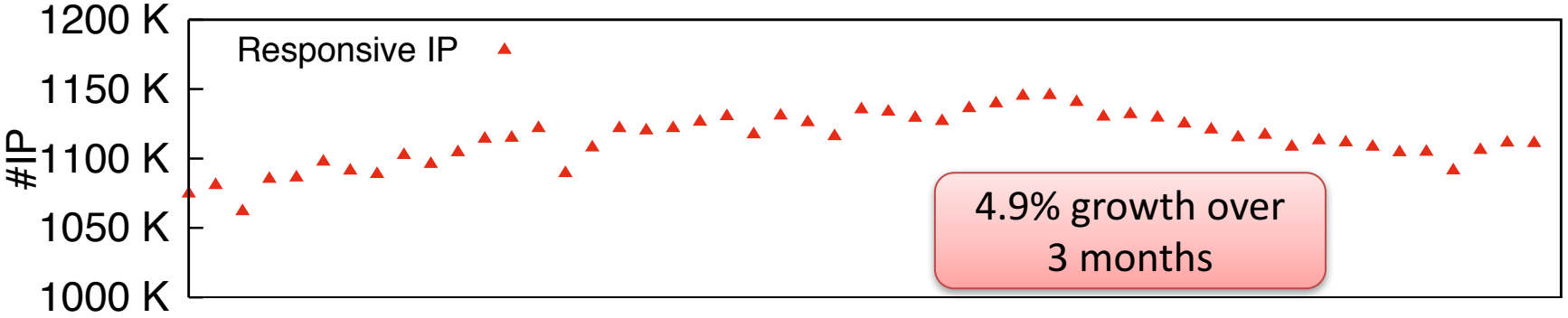
The ongoing migration to the cloud

HTTP request

Mean #IP/Round
33,145
5,597
2,029
1,167
617



Amazon EC2: 4,702,208 target IP addresses



2012: 4% of Alexa Top 1 Million domains make use of EC2/Azure

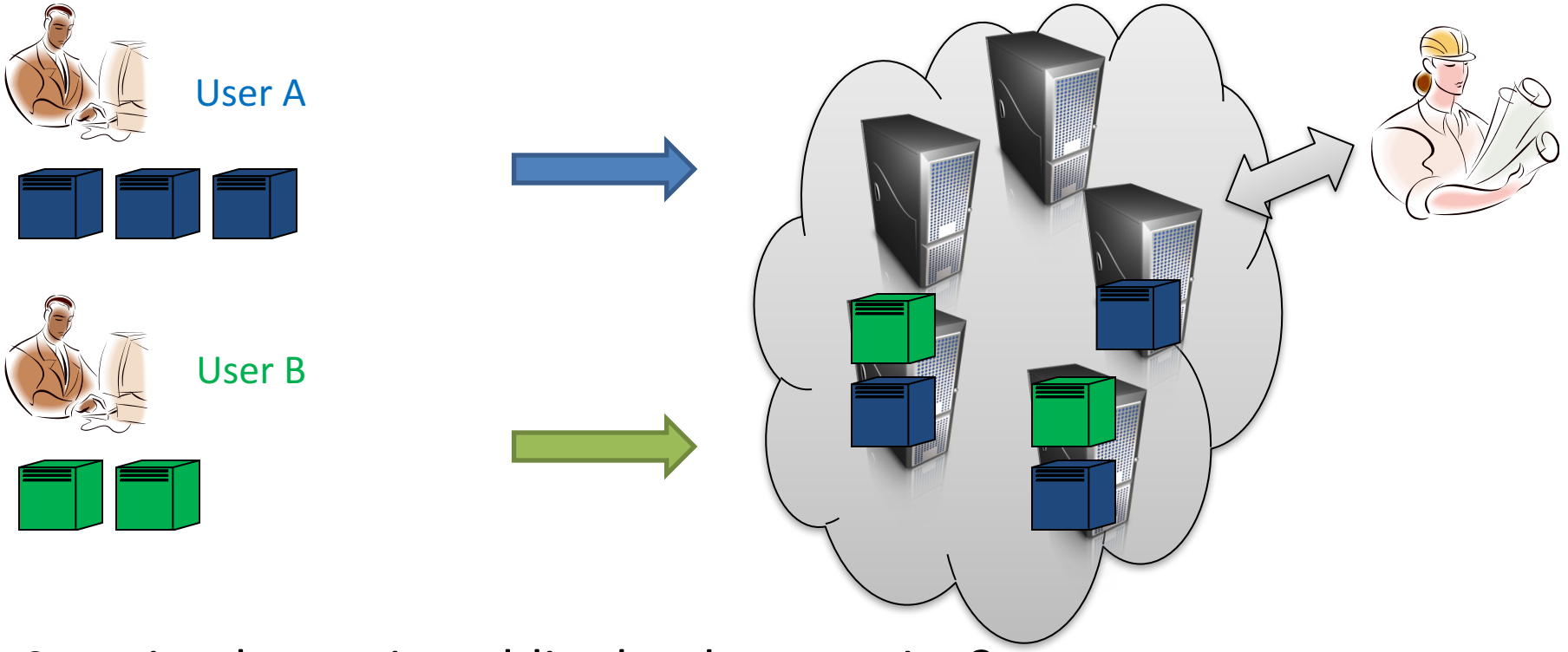
[He, Wang, Fisher, Gembler, Akella, R. – IMC 13] [Wang, Nappa, Caballero, R., Akella – IMC 14]

If you're having trouble accessing several parts of the Web today, it's because Amazon Web Services is having a bit of a glitch this morning.

A server issue in Virginia is affecting most of the northeast, killing the infrastructure for many popular products and services including Netflix, Product Hunt, Medium, SocialFlow, Buffer, GroupMe, Pocket, Viber Amazon Echo and more. The outage occurred around 6 AM ET.

http://thenextweb.com/insider/2015/09/20/amazon-web-services-goes-down-taking-netflix-reddit-pocket-and-more-with-it/

Trust models in public cloud computing



Security threats in public cloud computing?

External attacks against infrastructure

Adversarial provider spying on running VMs / data

Cross-user attacks

- VM image attacks
- Side-channels attacks

Private Medical Data of Over 1.5 Million People Exposed Through Amazon

94

Posted by **timothy** on Saturday September 19, 2015 @11:32PM from the pretty-soon-you're-talking-real-people dept.

Gizmodo reports that a wide variety of information about 1.5 million people -- everything from police injury reports, doctor's notes about their patients, and social security numbers -- "all were inexplicably unveiled on a public subdomain of Amazon Web Services. Welcome to the next big data breach horrorshow. Instead of hackers, it's old-fashioned neglect that exposed your most sensitive information." From the article:

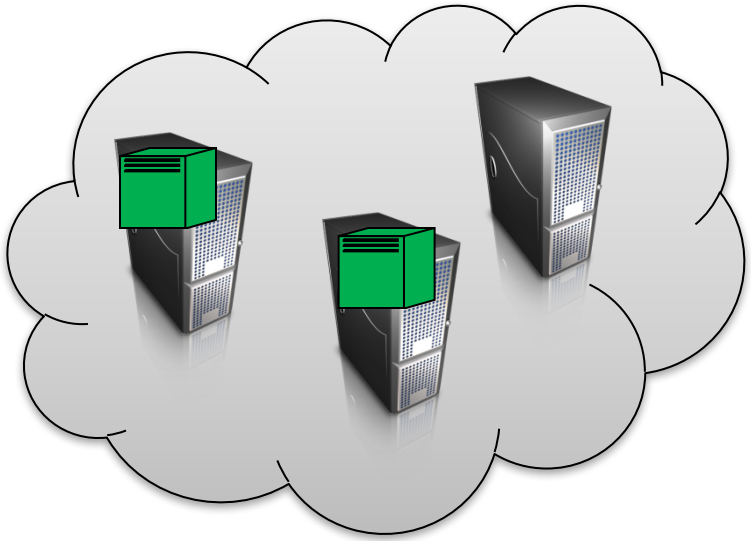
> Tomorrow, [Texas-based researcher Chris Vickers, who discovered the breach] will turn over the data to the the Texas Attorney General, where it will be destroyed. But that doesn't mean Systema is in the clear. Vickers may not be the only person who downloaded those millions of records as they sat out in the Amazon cloud. We don't know how long the information was available for everyone to see. But no matter what the timeframe, the neglect could be a HIPAA violation: Systema failed to protect the security of patients' electronic medical information.

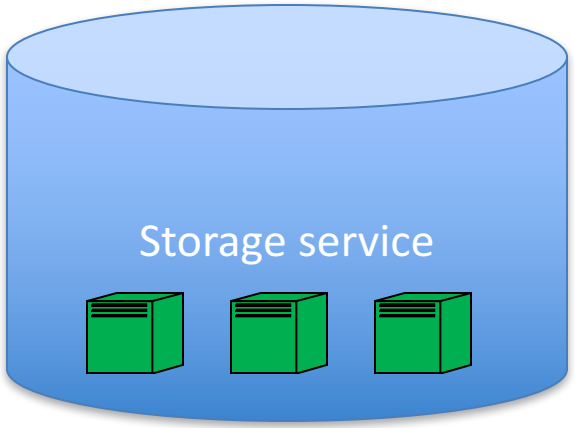
Virtual Machine Management

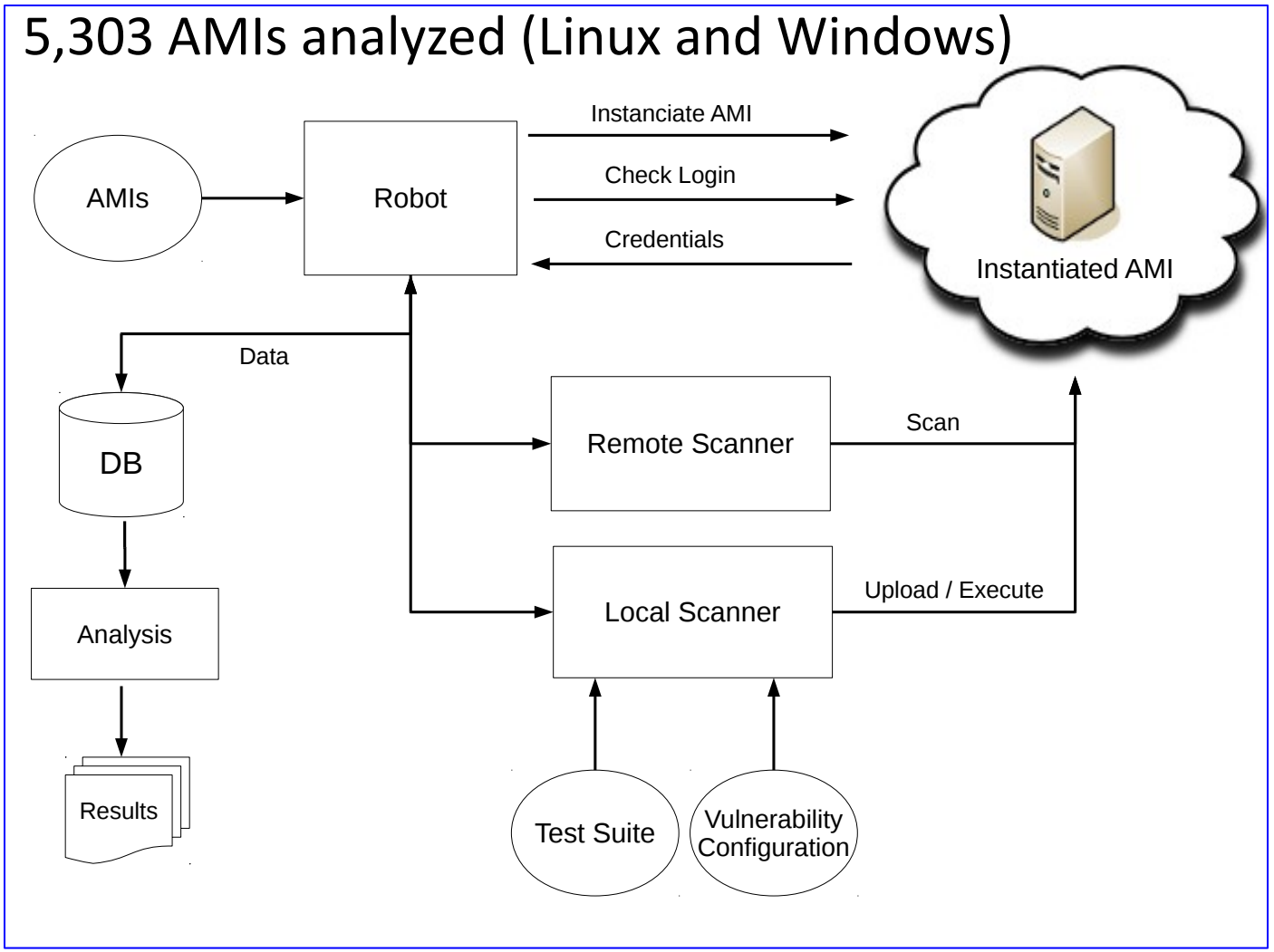
- Snapshots
  - Volume snapshot / checkpoint
    - persistent storage of VM
    - must boot from storage when resuming snapshot
  - Full snapshot
    - persistent storage and ephemeral storage (memory, register states, caches, etc.)
    - start/resume in between (essentially) arbitrary instructions
- VM image is a file that stores a snapshot

Amazon Machine Images (AMIs)

- Users set up volume snapshots / checkpoints that can then be run on the Elastic Compute Cloud (EC2)
- Can be marked as public and anyone can use your AMI

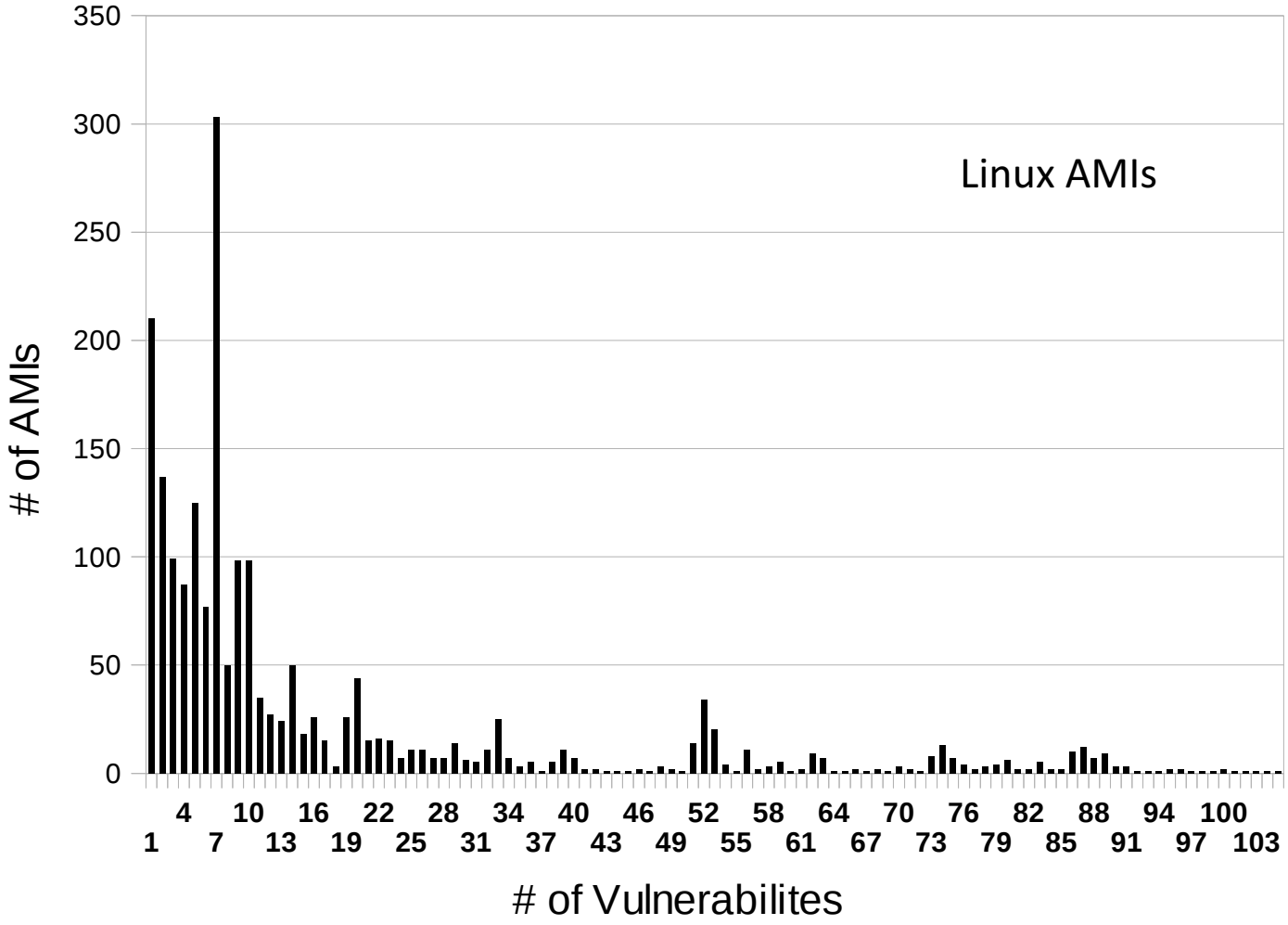






Balduzzi et al. “A Security Analysis of Amazon’s Elastic Compute Cloud Service – Long Version –”, 2011

See also Bugiel et al., “AmazonIA: When Elasticity Snaps Back”, 2011



Also: Malware found on a couple AMIs

Balduzzi et al. “A Security Analysis of Amazon’s Elastic Compute Cloud Service – Long Version –”, 2011

Balduzzi et al. analysis

- Backdoors
  - AMIs include SSH public keys within authorized_keys
  - Password-based backdoors

	East	West	EU	Asia	Total
AMIs (%)	34.8	8.4	9.8	6.3	21.8
With Passwd	67	10	22	2	101
With SSH keys	794	53	86	32	965
With Both	71	6	9	4	90
Superuser Priv.	783	57	105	26	971
User Priv.	149	12	12	12	185

Table 2: Left credentials per AMI

Balduzzi et al. analysis

- Credentials for other systems
  - AWS secret keys (to control EC2 services of an account): 67 found
  - Passwords / secret keys for other systems: 56 found

Finding	Total	Image	Remote
Amazon RDS	4	0	4
dDNS	1	0	1
SQL	7	6	1
MySql	58	45	13
WebApp	3	2	1
VNC	1	1	0
Total	74	54	20

Table 3: Credentials in history files

Balduzzi et al. analysis

- Deleted files
  - One AMI creation method does block-level copying

Type	#
Home files (`/home, /root`)	33,011
Images (min. 800x600)	1,085
Microsoft Office documents	336
Amazon AWS certificates and access keys	293
SSH private keys	232
PGP/GPG private keys	151
PDF documents	141
Password file (`/etc/shadow`)	106

Table 5: Recovered data from deleted files

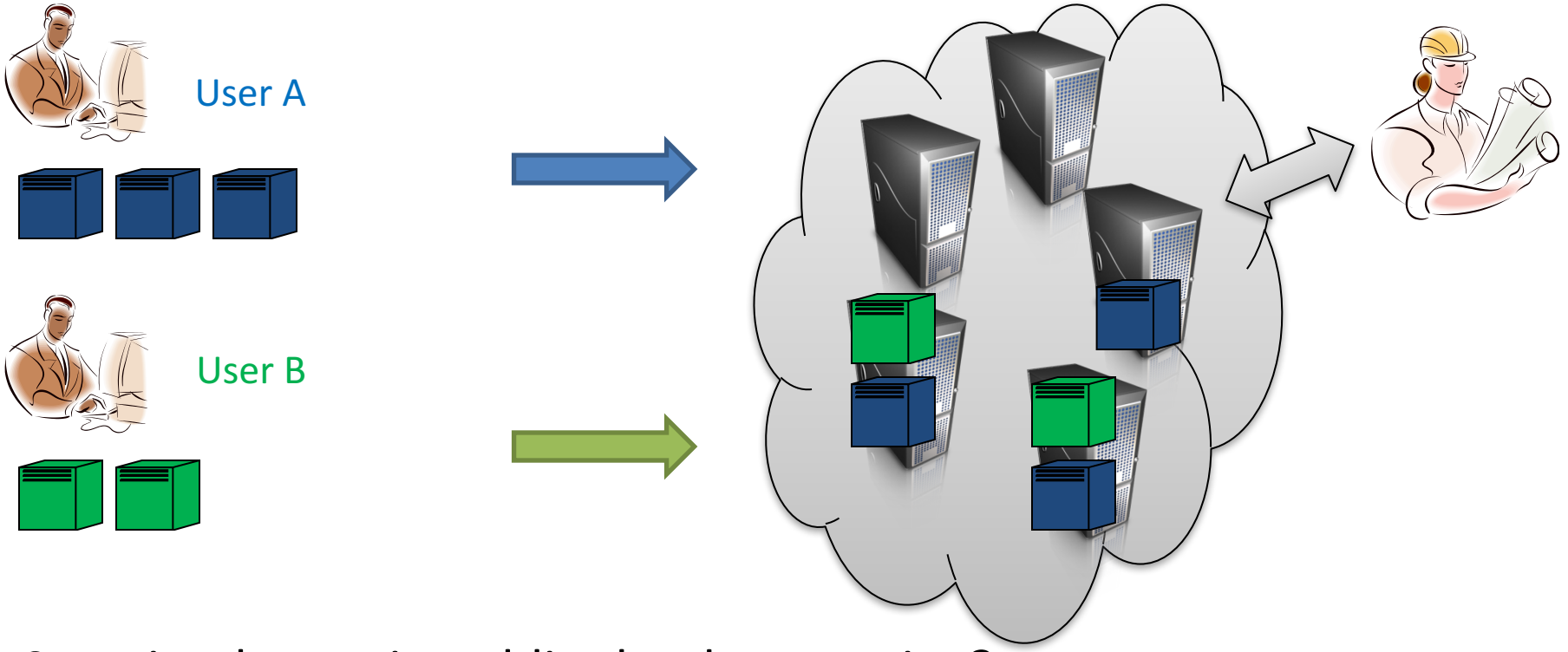
Response

"They told me it's not their concern, they just provide computing power," Balduzzi says. "It's like if you upload naked pictures to Facebook. It's not a good practice, but it's not Facebook's problem."

http://www.forbes.com/sites/andygreenberg/2011/11/08/researchers-find-amazon-cloud-servers-teeming-with-backdoors-and-other-peoples-data/

- Amazon notified customers with vulnerable AMIs
- Made private AMIs of non-responsive customers
- New tutorials for bundling systems
- Working on undelete issues…

Trust models in public cloud computing



Security threats in public cloud computing?

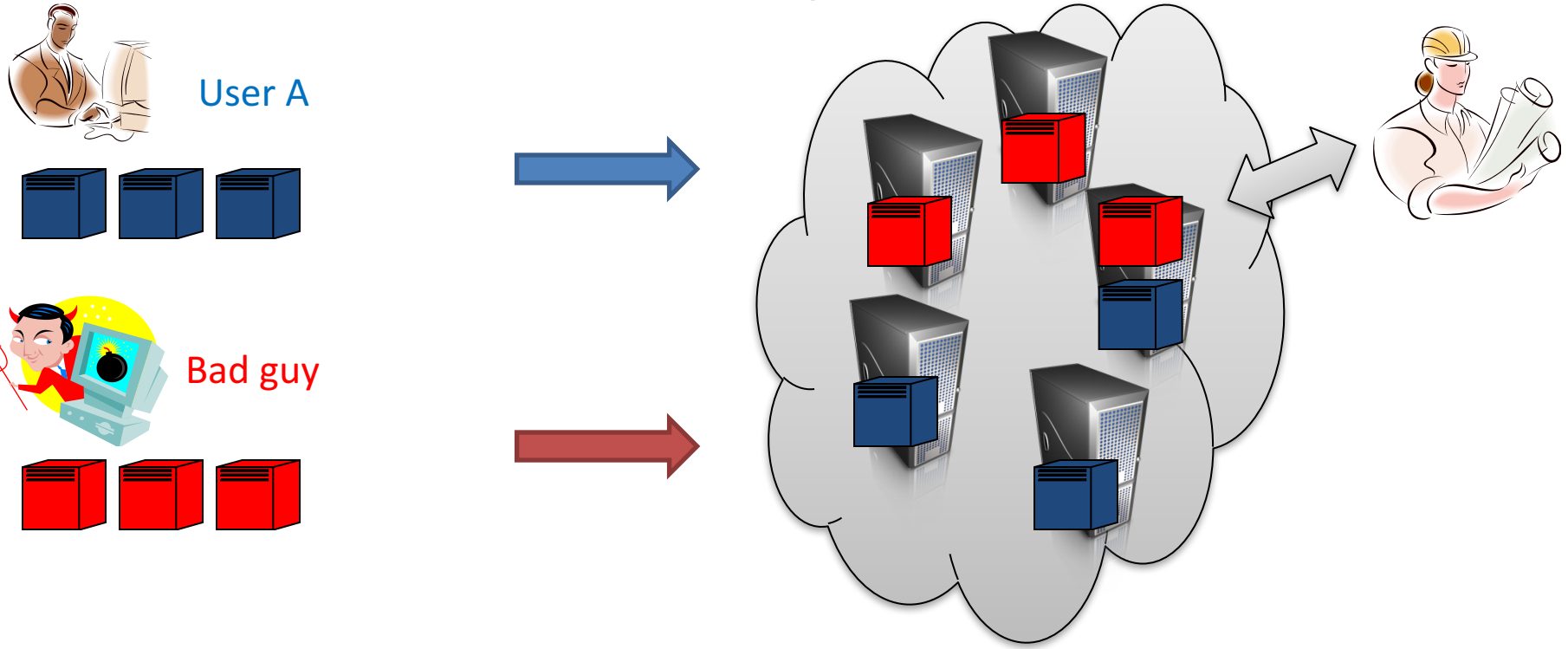
External attacks against infrastructure

Adversarial provider spying on running VMs / data

Cross-user attacks

- VM image attacks
- Side-channels attacks

Threats due to resource sharing



Attacker identifies one or more victims VMs in cloud

1) Achieve advantageous placement via launching of VM instances

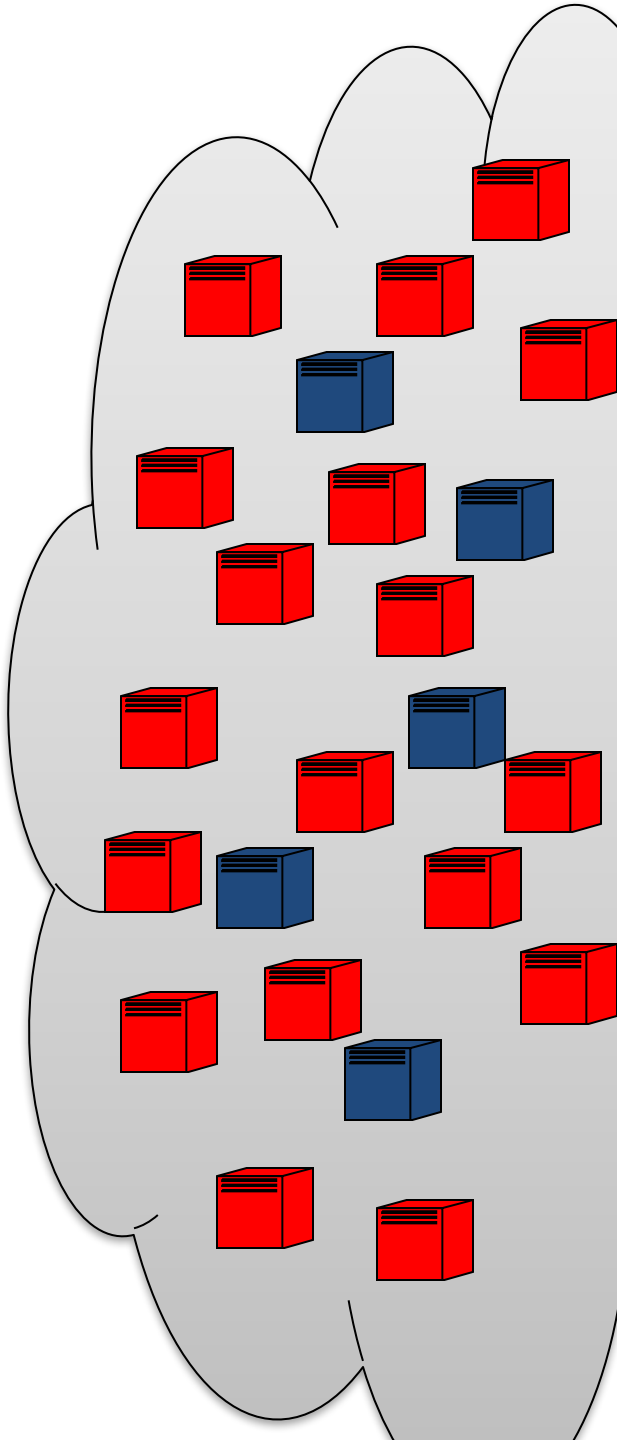
2) Launch attacks using physical proximity

Exploit VMM vulnerability DoS Side-channel attack

1 or more targets in the cloud and we want to attack them from same physical host

Launch lots of instances (over time), with each attempting an attack

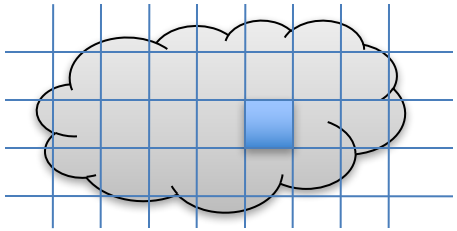
Can attackers do better?



Outline of a more damaging approach:

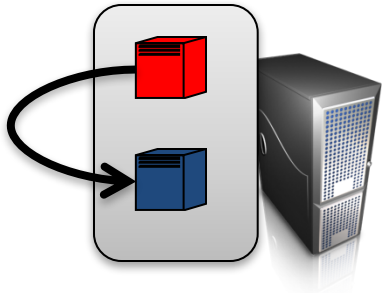
1) Cloud cartography

map internal infrastructure of cloud
map used to locate targets in cloud



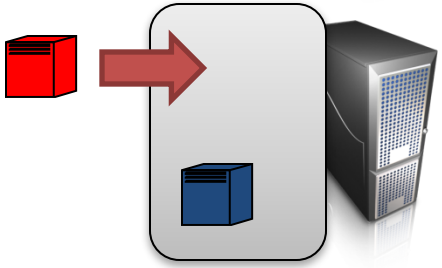
2) Checking for co-residence

check that VM is on same server as target
- network-based co-residence checks
- efficacy confirmed by covert channels



3) Achieving co-residence

brute forcing placement
instance flooding after target launches



4) Location-based attacks

side-channels, DoS, escape-from-VM

Placement vulnerability: attackers can knowingly achieve co-residence with target

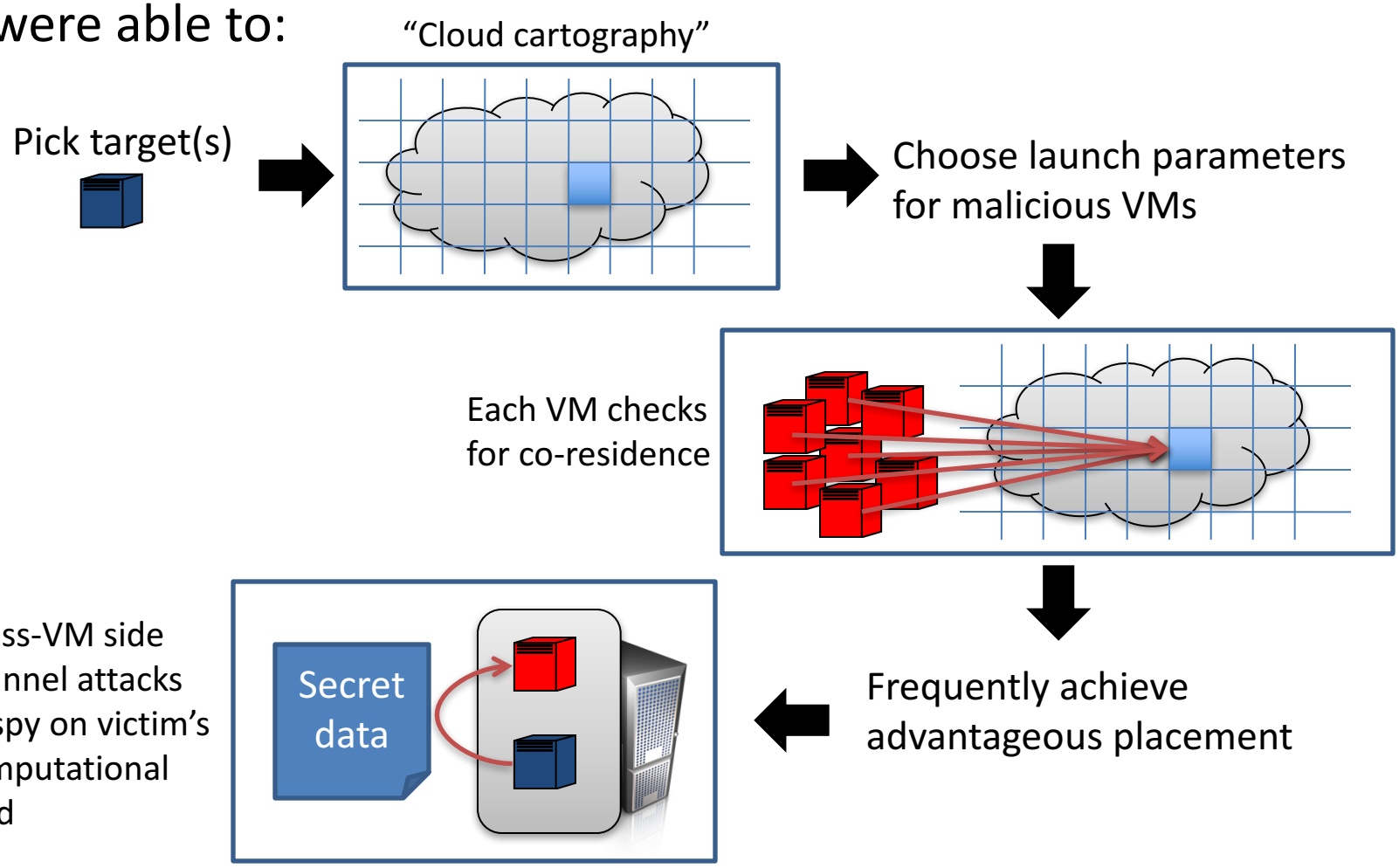
Case study with Amazon's EC2

1) given no insider information

2) restricted by (the spirit of) Amazon's acceptable use policy (AUP)

(using only Amazon's customer APIs and very restricted network probing)

We were able to:



Some info about EC2 service (at time of study)

Linux-based VMs available
Uses Xen-based VM manager

launch parameters

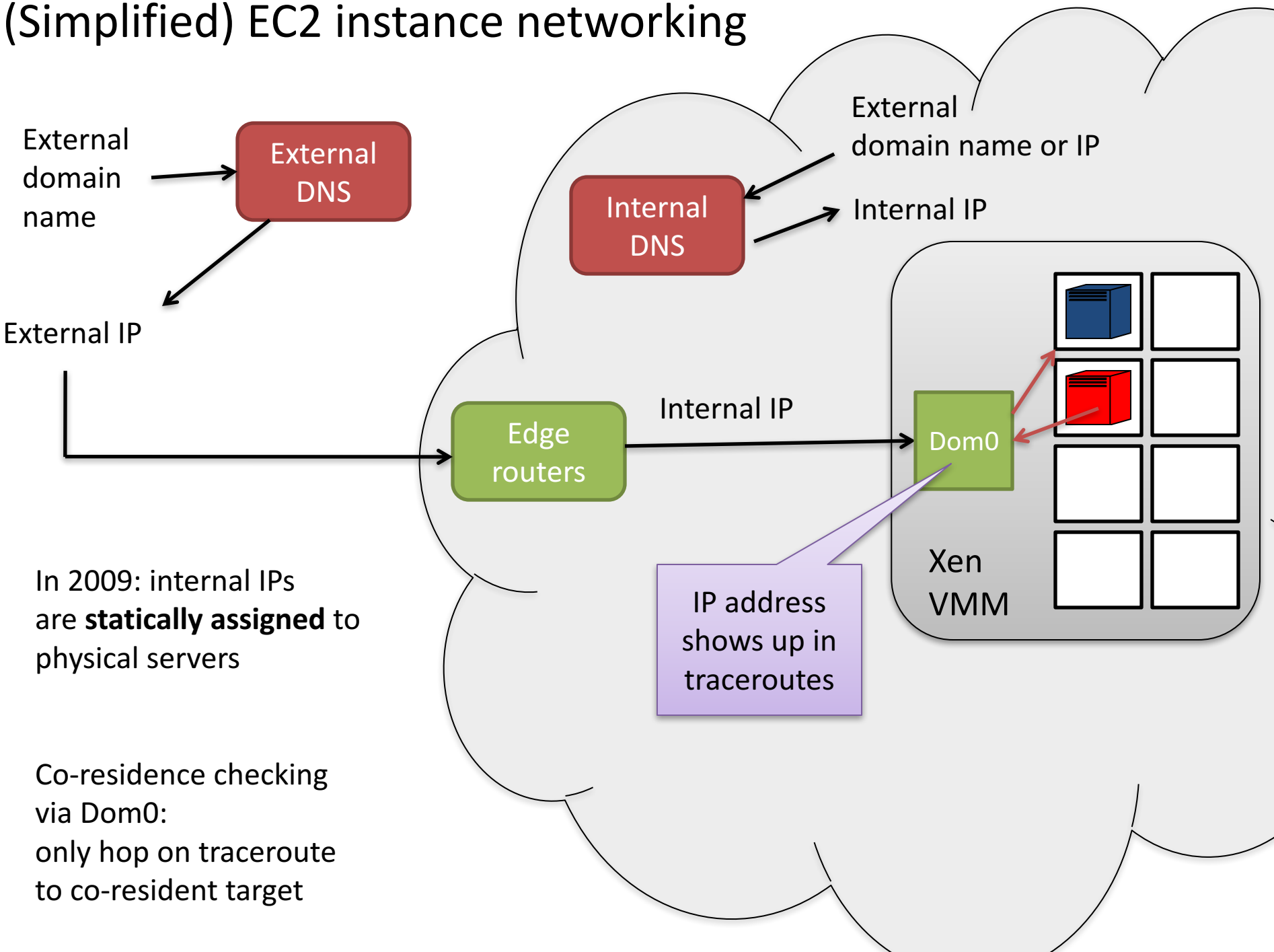
- User account
- 3 “availability zones” (Zone 1, Zone 2, Zone 3)
- 5 instance types (various combinations of virtualized resources)

Type	gigs of RAM	EC2 Compute Units (ECU)
m1.small (default)	1.7	1
m1.large	7.5	4
m1.xlarge	15	8
c1.medium	1.7	5
c1.xlarge	7	20

1 ECU = 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor

Limit of 20 instances at a time per account.
Essentially unlimited accounts with credit card.

(Simplified) EC2 instance networking



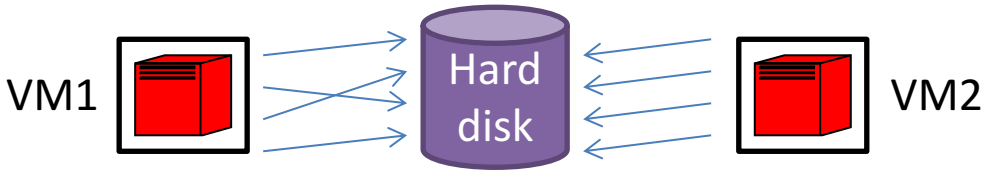
In 2009: internal IPs are **statically assigned** to physical servers

Co-residence checking via Dom0:
only hop on traceroute to co-resident target

Checking for co-residence: Covert channels

Covert channels allow cooperating sender and receiver to send a message to each other while skirting information flow controls

Lampson. A Note on the Confinement Problem. 1973



Sender transmits ‘1’ by franticly reading random locations

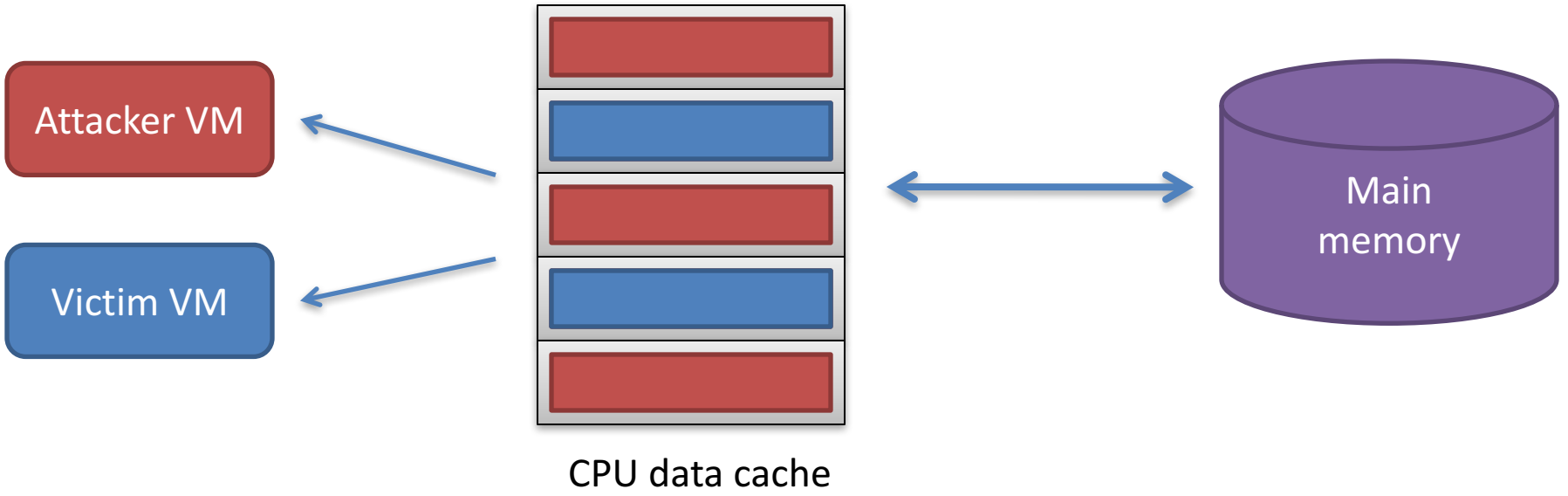
Sender transmits ‘0’ by doing nothing

Receiver times reading of a fixed location

Covert channels require control of both VMs:
Not directly useful to determine co-residency with target victim

Checking for co-residence: Side channels

A side-channel is an unintentional leakage of information about confidential process

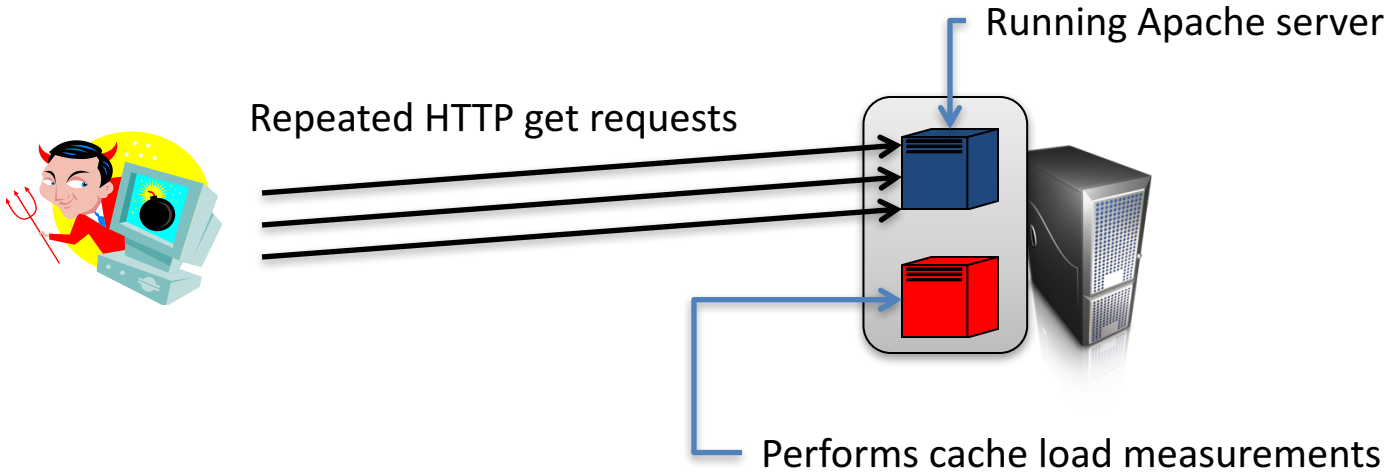


1) Read in a large array (fill CPU cache with attacker data)

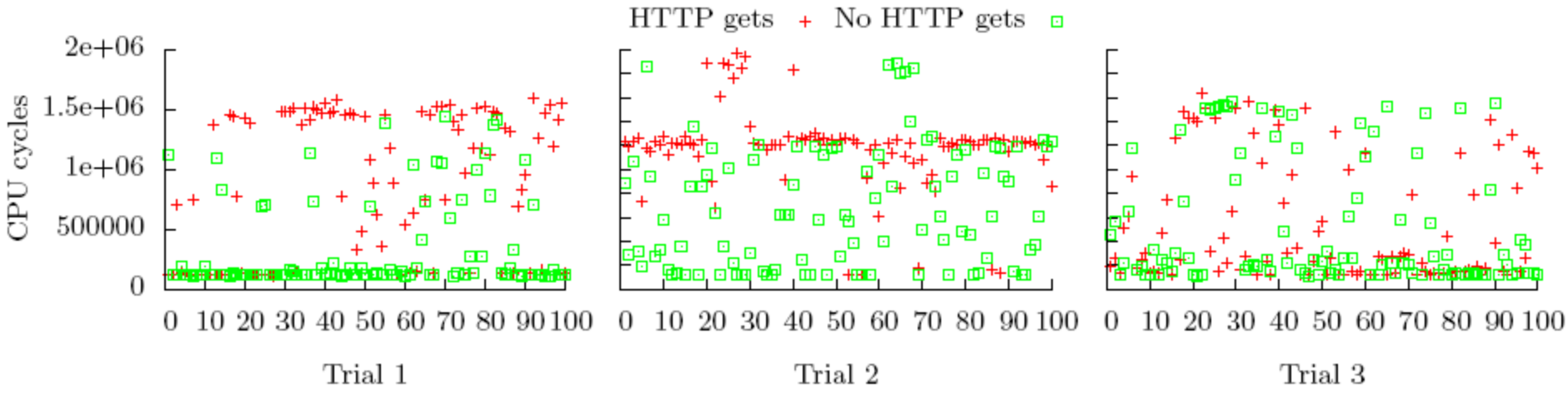
2) Busy loop (allow victim to run)

3) Measure time to read large array (the *load measurement*)

Cache-based cross-VM load measurement on EC2



3 pairs of instances, 2 pairs co-resident and 1 not
100 cache load measurements during **HTTP gets** (1024 byte page) and with **no HTTP gets**



Instances co-resident | Instances co-resident | Instances NOT co-resident

Checking for co-residence

Experiment

Repeat 3 times:

1) 20 m1.small Account A
2) 20 m1.small Account B
3) All pairs w/ matching Dom0 → send 5-bit message across HD covert channel

Dom0 check works

Ended up with 31 pairs of co-resident instances as indicated by Dom0 IPs

Result: a correctly-received message sent for every pair of instances

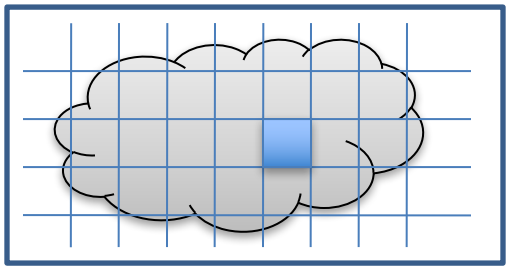
During experiment also performed pings to:
* 2 control instances in each zone
* co-resident VM

RTT times also indicate co-residence

	Median RTT (ms)
Zone 1 Control 1	1.164
Zone 1 Control 2	1.027
Zone 2 Control 1	1.113
Zone 2 Control 2	1.187
Zone 3 Control 1	0.550
Zone 3 Control 2	0.436
Co-resident VM	0.242

Cloud cartography

Pick target(s)



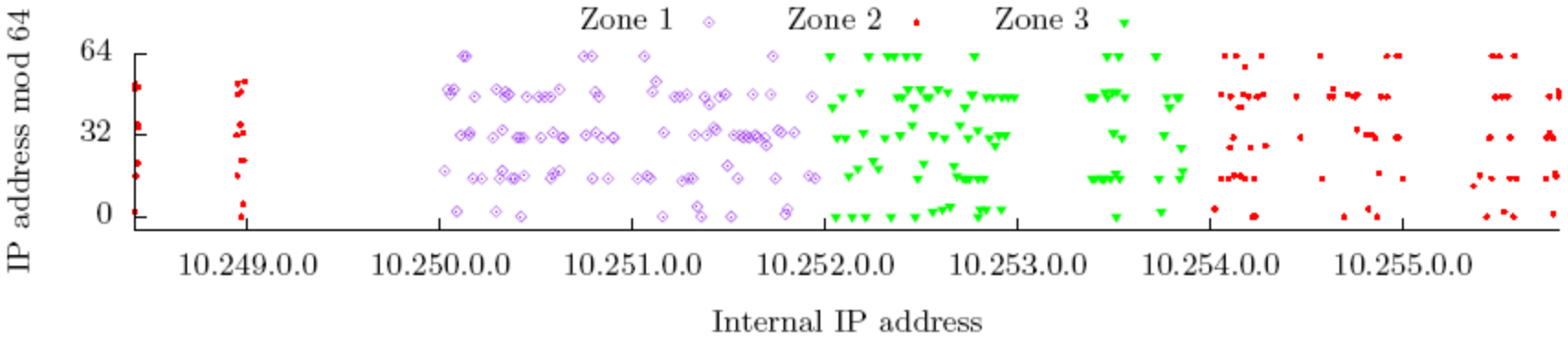
Choose launch parameters for malicious VMs

launch parameters

3 "availability zones"
(Zone 1, Zone 2, Zone 3)

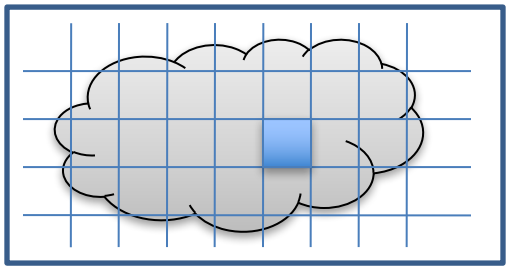
5 instance types
(m1.small, c1.medium, m1.large, m1.xlarge, c1.xlarge)

User account



Cloud cartography

Pick target(s)



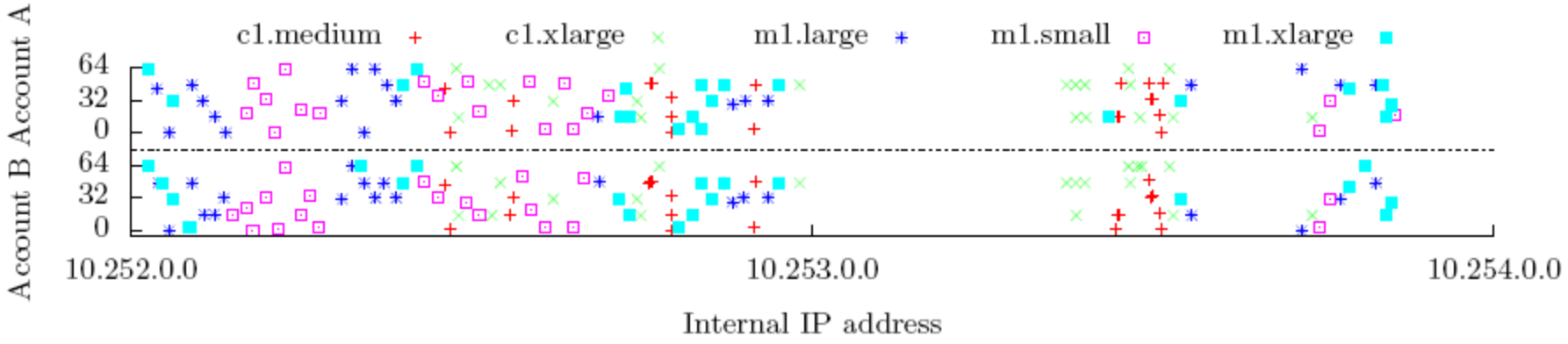
Choose launch parameters for malicious VMs

launch parameters

3 "availability zones"
(Zone 1, Zone 2, Zone 3)

5 instance types
(m1.small, c1.medium, m1.large, m1.xlarge, c1.xlarge)

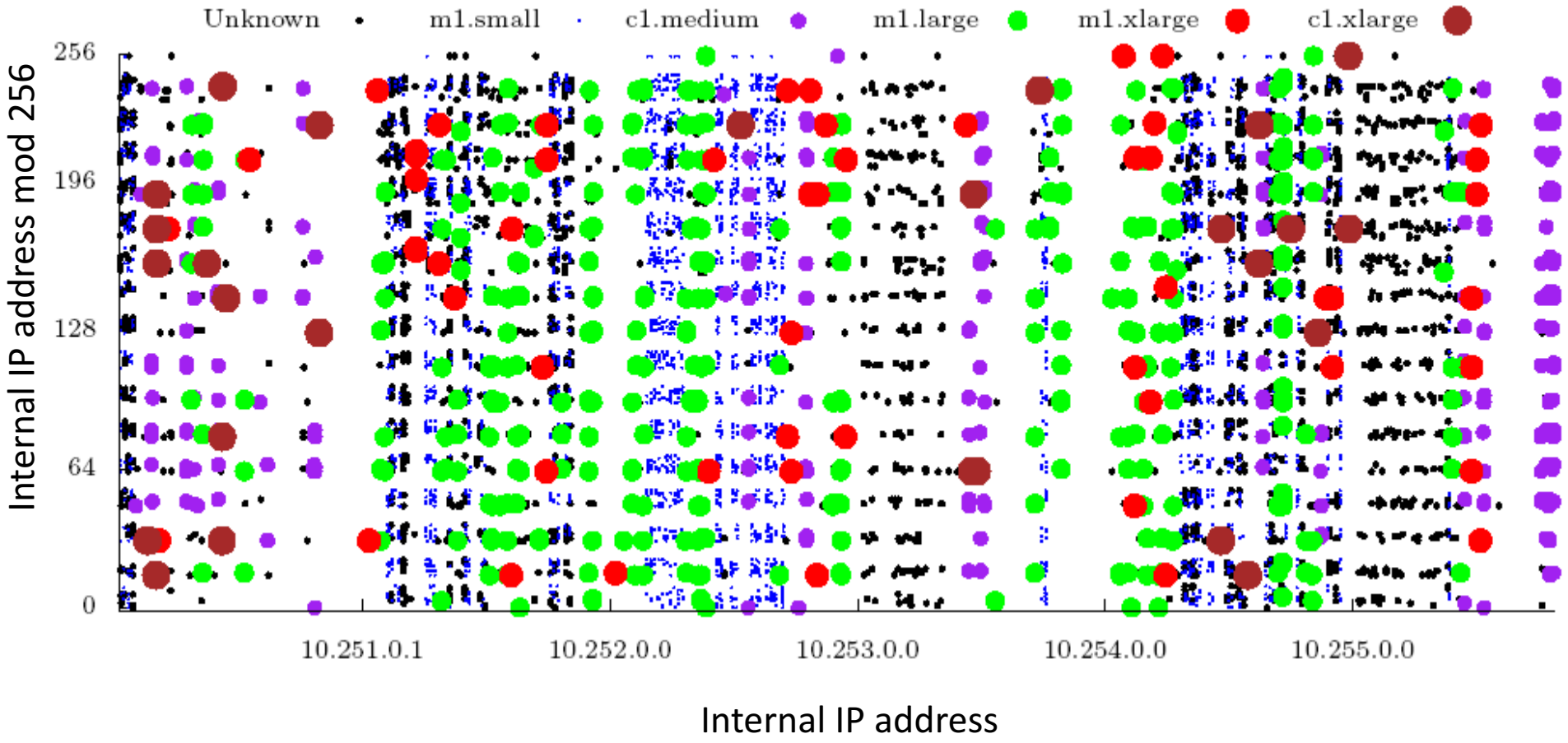
User account



Associate to each /24 an estimate of Availability zone and Instance Type

External IP —DNS→ Internal IP —/24→ Availability zone, Instance Type

Mapping 6,577 public HTTP servers running on EC2 (Fall 2008)



Achieving co-residence

"Brute-forcing" co-residence

Attacker launches many VMs over a relatively long period of time in target's zone and of target type

Experiment:

1,686 public HTTP servers as stand-in "targets" running m1.small and in Zone 3 (via our map)

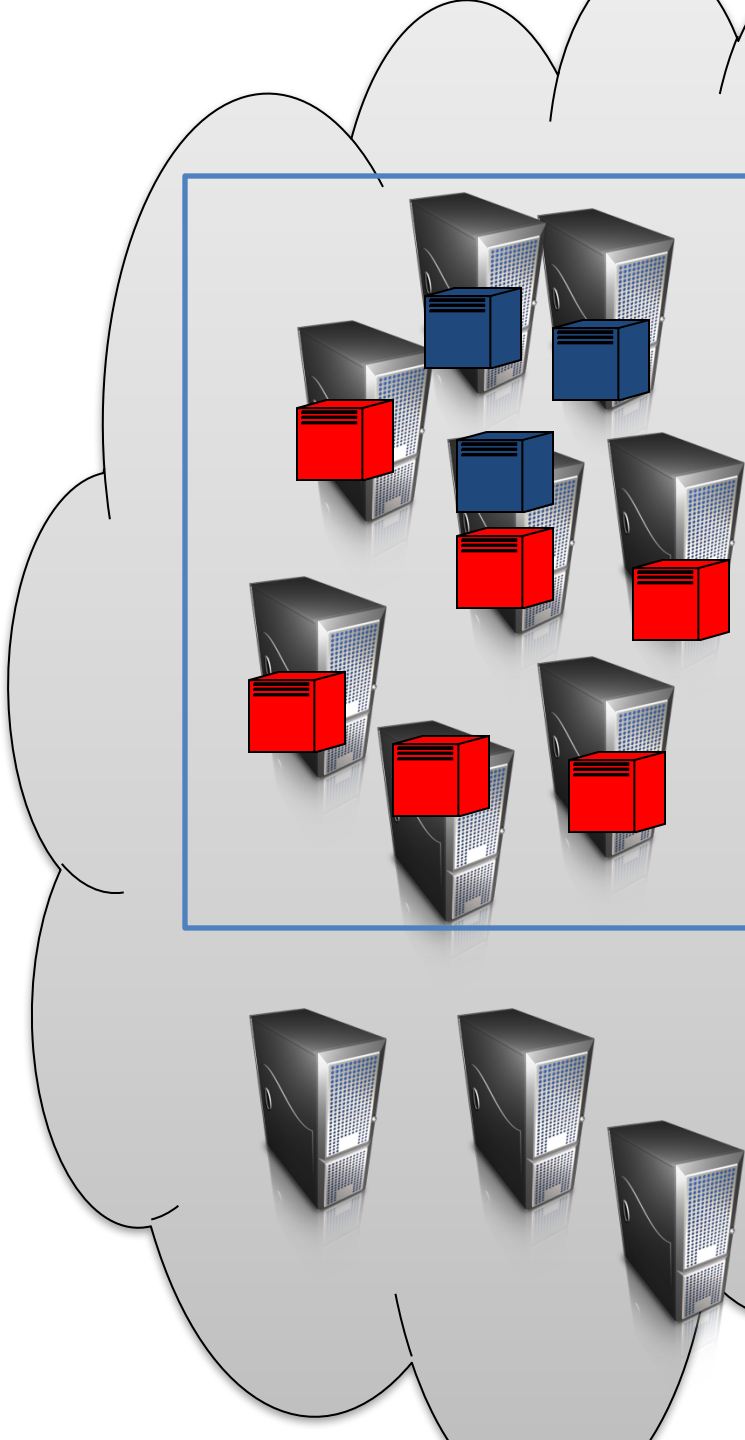
1,785 "attacker" instances launched over 18 days

Each checked co-residence against all targets using Dom0 IP

Results:

78 unique Dom0 IPs

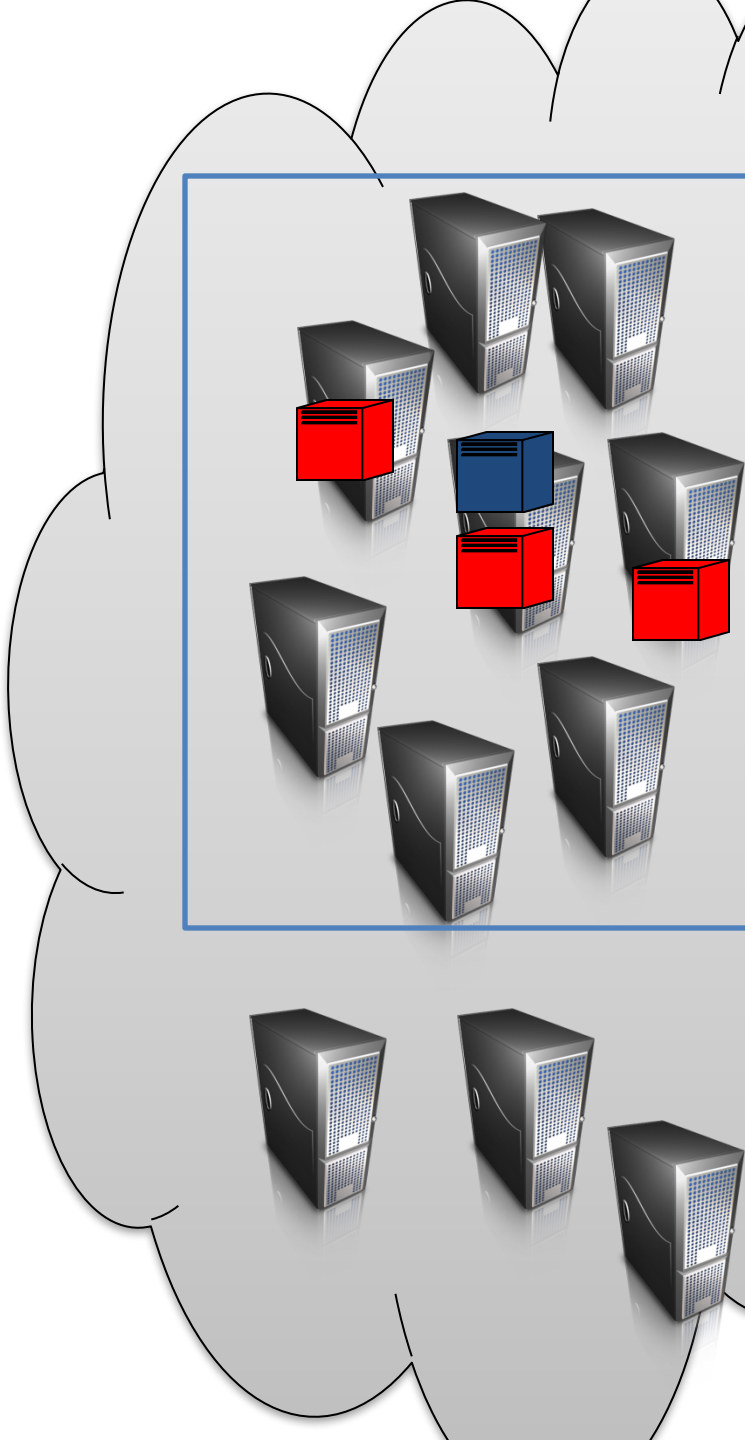
141 / 1,686 (8.4%) had attacker co-resident



Achieving co-residence

Instance flooding near target launch abuses parallel placement locality

Launch many instances in parallel into known zone/type near time of target launch



Experiment:

Repeat for 10 trials:

1) Launch 1 target VM (Account A)
2) 5 minutes later, launch 20 “attack” VMs (alternate using Account B or C)
3) Determine if any co-resident with target using Dom0 IP

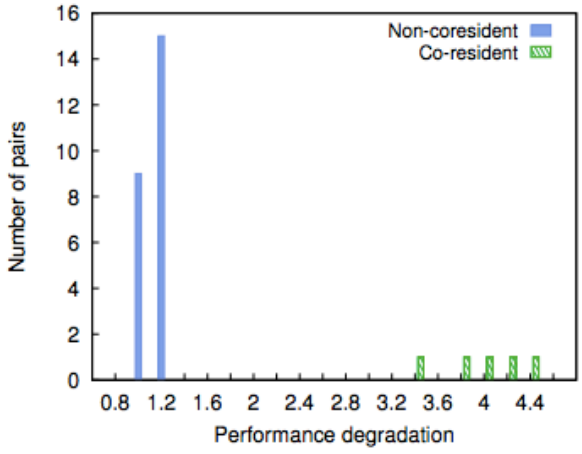
4 / 10 trials succeeded

Changes since 2009

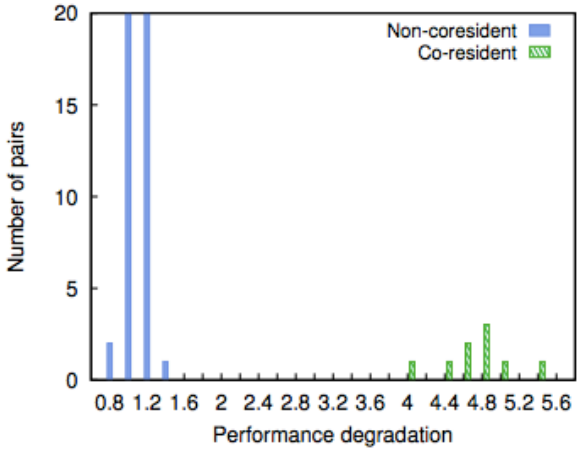
- Turned off Dom0
- Virtual Private Clouds
  - Instances launched into virtual private networks
  - Logically isolated namespace
    - Prevents cloud cartography based on internal IPs
- Scale significantly larger
  - Liu 2013: ~450,000 servers for EC2

Newer placement studies

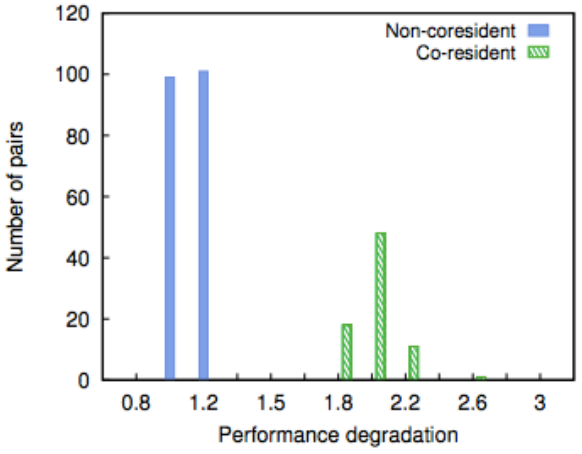
- Varadarajan et al. and Xu et al. (USENIX 2015)
- New co-residence tests, explore larger variety of settings
- Conclusions: can still achieve co-residency



(a) GCE

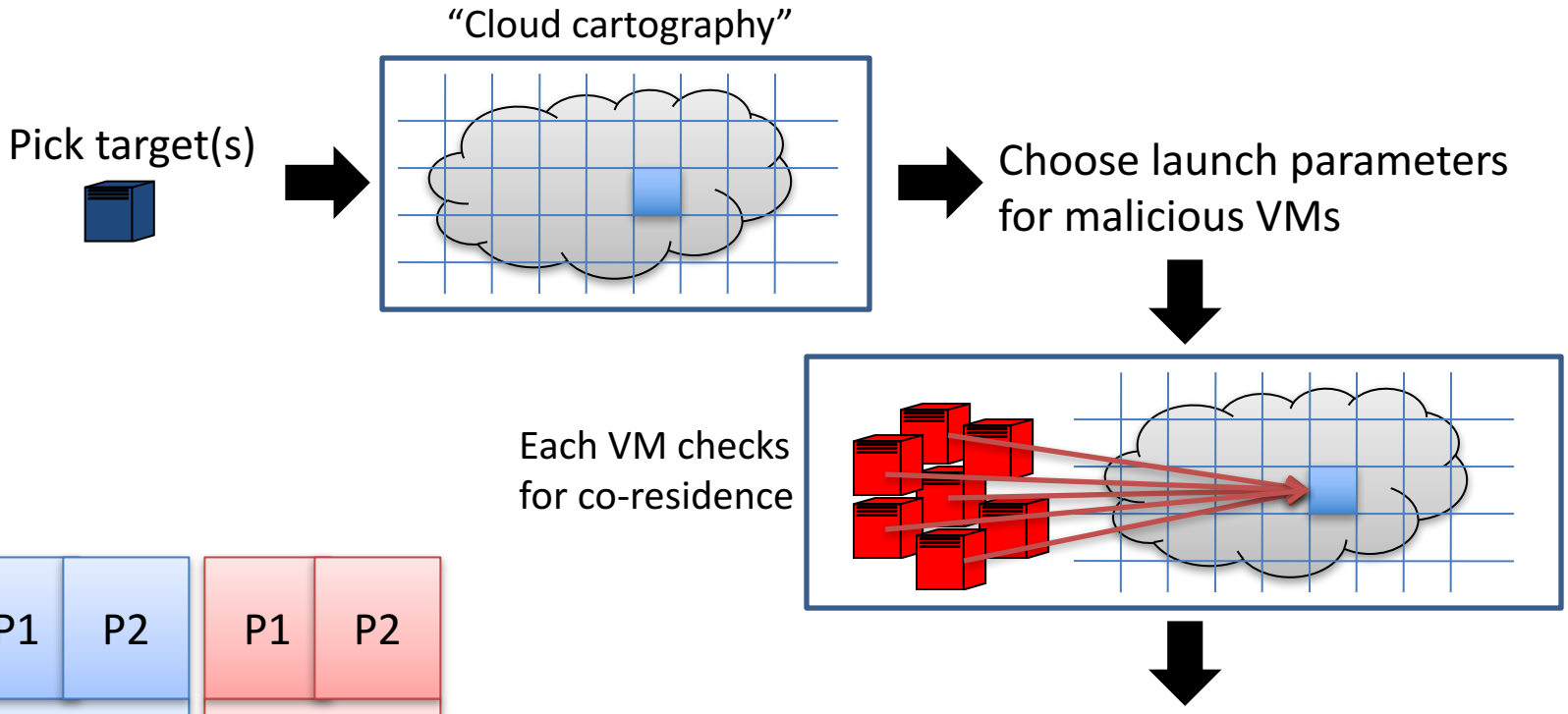


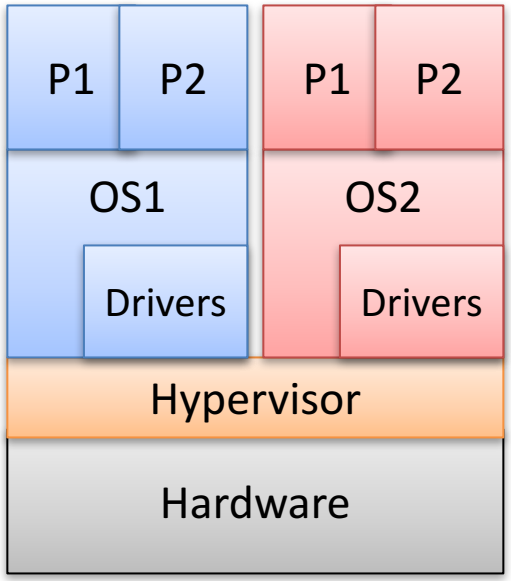
(b) EC2



(c) Azure – Intel machines

So far we were able to:

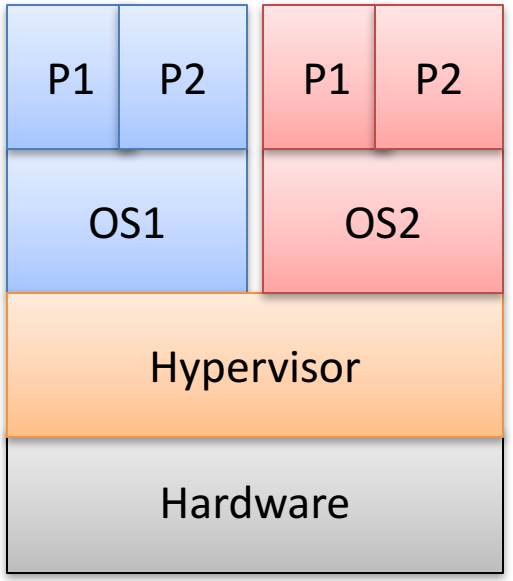




This shouldn't matter if VMM provides good isolation!

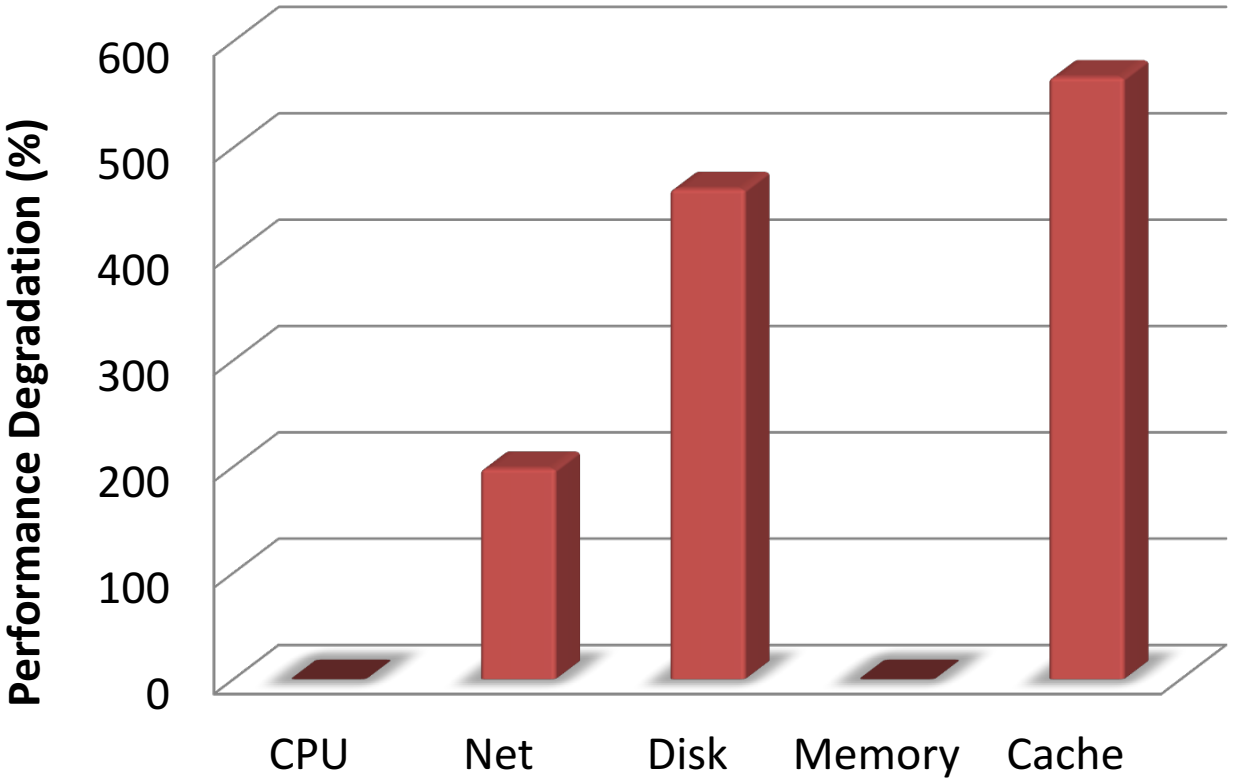
Violating isolation

- Cross-VM side-channel attacks
- Degradation-of-Service attacks
  - Guests might maliciously contend for resources
  - Xen scheduler vulnerability
- Escape-from-VM vulnerabilities
- Resource-freeing attacks



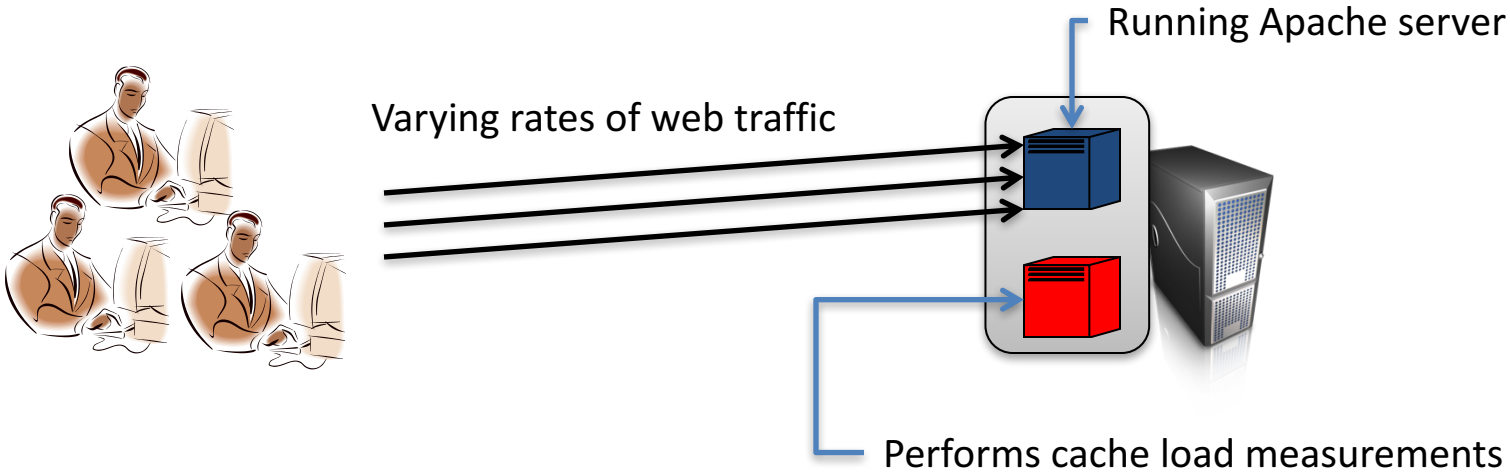
Measuring Resource Contention

- Contention for the same resource

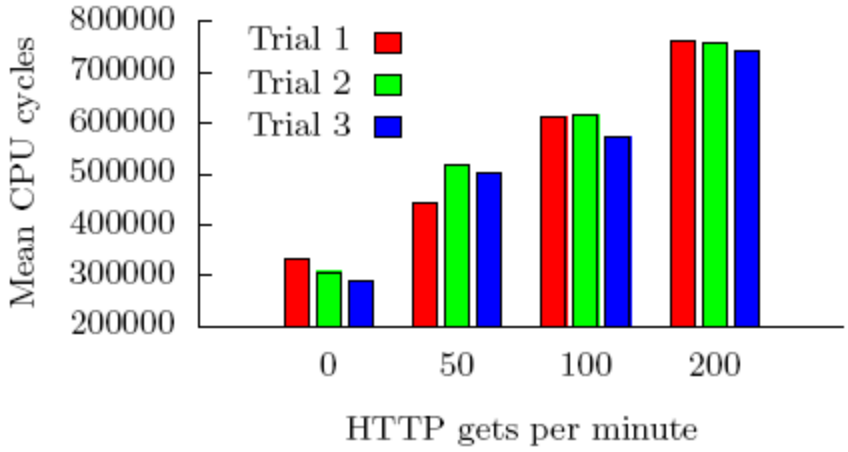


Local Xen Testbed	
Machine	Intel Xeon E5430, 2.66 Ghz
Packages	2, 2 cores per package
LLC Size	6MB per package

Cache-based load measurement of traffic rates on EC2



3 trials with 1 pair of co-resident instances:
1000 cache load measurements during
0, 50, 100, or 200 **HTTP gets** (3 Mbyte page) per minute for ~1.5 mins



What can cloud providers do?

1) Clo

2) Che… co-re…

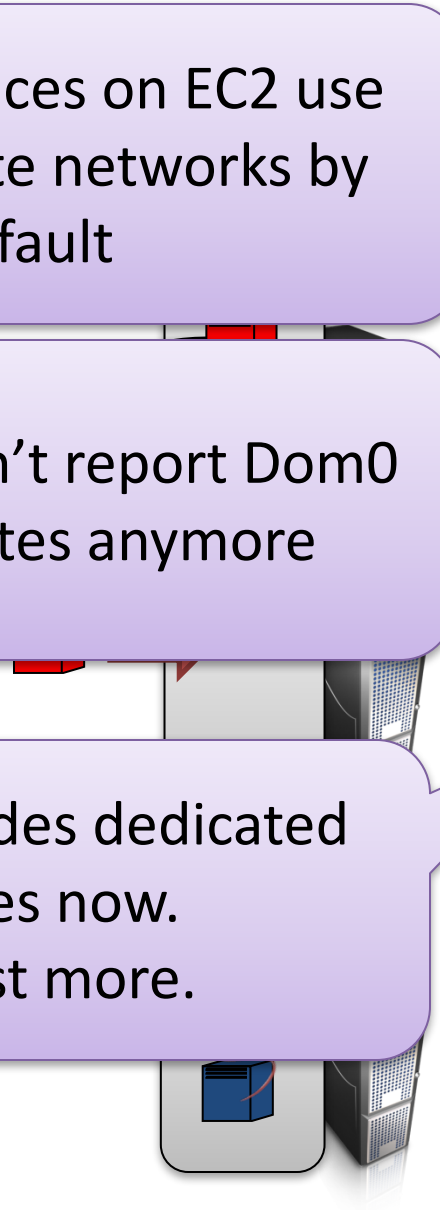
3) Ac… co-residence

4) Si… infor…

Newer instances on EC2 use virtual private networks by default

Amazon doesn't report Dom0 in traceroutes anymore

Amazon provides dedicated instances now.
They cost more.



Possible counter-measures:

- Random Internal IP assignment
- Isolate each user's view of internal address space

- Hide Dom0 from traceroutes

- Allow users to opt out of multitenancy

- Hardware or software countermeasures to stop leakage [Ber05,OST05,Page02,Page03, Page05,Per05]
- Improved performance isolation

Next time:
Side-channel attacks

- Yarom & Falkner Flush+Reload attack on RSA
- Meltdown

What can cloud providers do?

1) Clo

2) Che… co-re…

3) Ac… co-residence

4) Si… infor…

Resource-freeing attacks

Newer instances on EC2 use virtual private networks by default

Amazon doesn't report Dom0 in traceroutes anymore

Amazon provides dedicated instances now.
They cost a lot more.

Possible counter-measures:

- Random Internal IP assignment
- Isolate each user's view of internal address space

- Hide Dom0 from traceroutes

- Allow users to opt out of multitenancy

- Hardware or software countermeasures to stop leakage [Ber05,OST05,Page02,Page03, Page05,Per05]
- Improved performance isolation

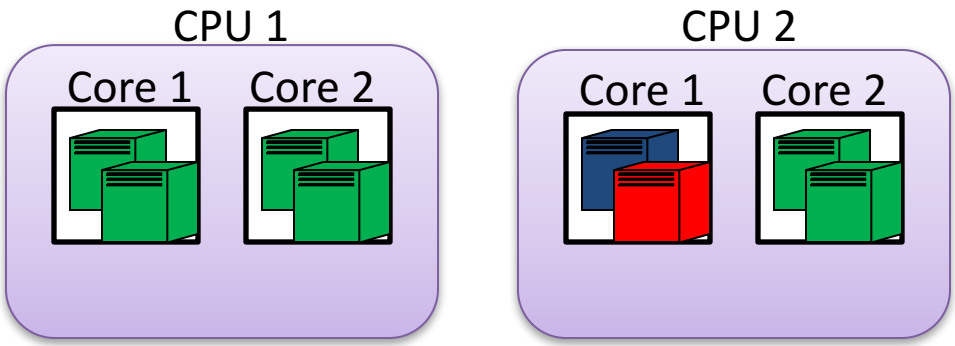
More on cache-based physical channels

Prime+Trigger+Probe combined with differential encoding technique gives high bandwidth cross-VM covert channel on EC2

See [Xu et al., "An Exploration of L2 Cache Covert Channels in Virtualized Environments", CCSW 2011]

Keystroke timing in experimental testbed similar to EC2 m1.small instances

AMD Opterons



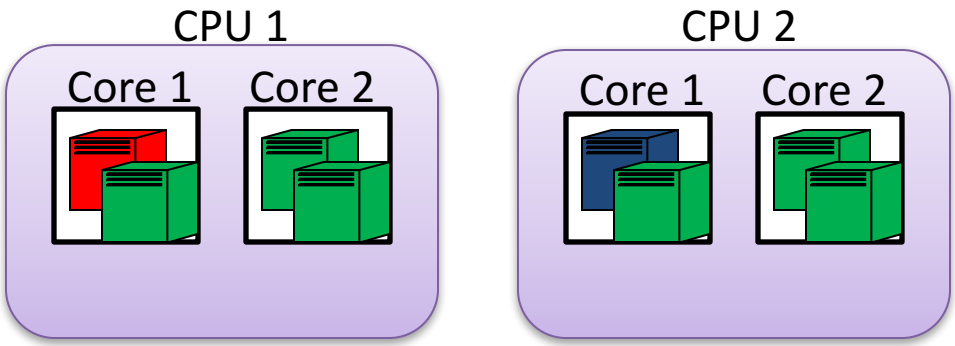
More on cache-based physical channels

Prime+Trigger+Probe combined with differential encoding technique gives high bandwidth cross-VM covert channel on EC2

See [Xu et al., "An Exploration of L2 Cache Covert Channels in Virtualized Environments", CCSW 2011]

Keystroke timing in experimental testbed similar to EC2 m1.small instances

AMD Opterons



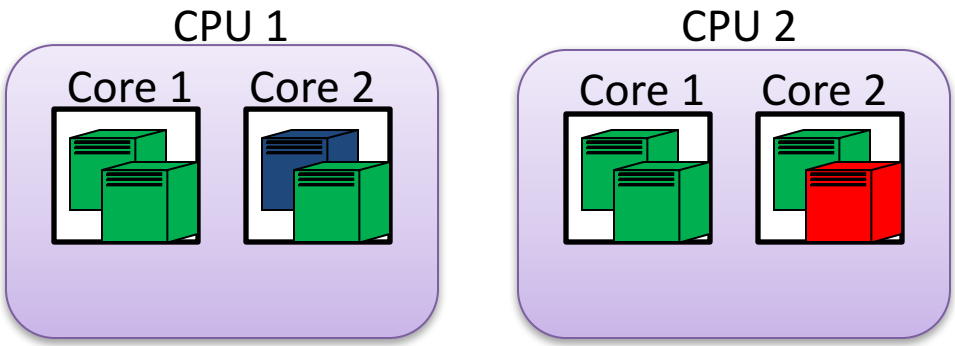
More on cache-based physical channels

Prime+Trigger+Probe combined with differential encoding technique gives high bandwidth cross-VM covert channel on EC2

See [Xu et al., "An Exploration of L2 Cache Covert Channels in Virtualized Environments", CCSW 2011]

Keystroke timing in experimental testbed similar to EC2 m1.small instances

AMD Opterons



More on cache-based physical channels

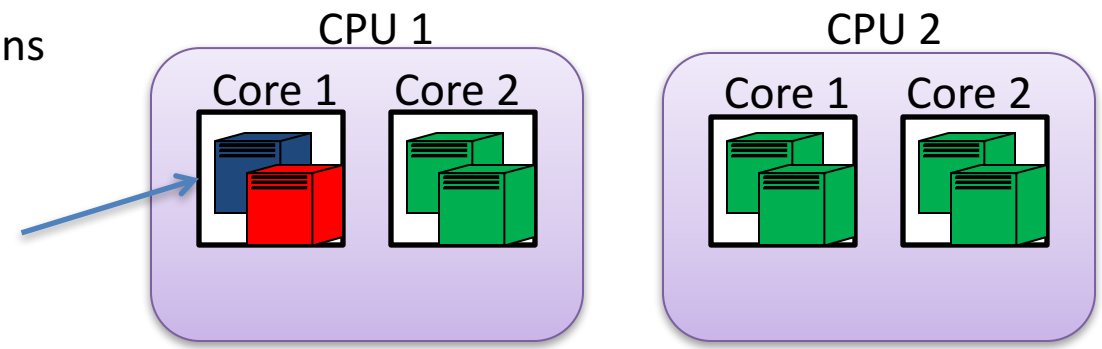
Prime+Trigger+Probe combined with differential encoding technique gives high bandwidth cross-VM covert channel on EC2

See [Xu et al., "An Exploration of L2 Cache Covert Channels in Virtualized Environments", CCSW 2011]

Keystroke timing in experimental testbed similar to EC2 m1.small instances

AMD Opterons

VMs pinned to core



We show that cache-load measurements enable cross-VM keystroke detection

Keystroke timing of this form might be sufficient for the password recovery attacks of [Song, Wagner, Tian 01]